

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ**

имени академика М.Д. Миллионщикова

Документ подписан простой электронной подписью

Информация о владельце:

Имя: Шинцаев Магомед Шавалович

Должность: Ректор

Дата подписания: 30.09.2023 16:03:54

Идентификационный программный ключ:

35c296f119d6aafdc22836b21db52dbc07971a86865a5825f9fa4304cc

«УТВЕРЖДАЮ»

Первый проректор

И.Г.Гайрабеков

« 01 » 09 2023г.

РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«Программирование»

Направление подготовки

09.03.01 «Информатика и вычислительная техника»

Направленность (профиль)

«Информатика и вычислительная техника»

Квалификация

Бакалавр

: 2023

Грозный - 2023

1. Цели и задачи дисциплины

Цель преподавания дисциплины «Программирование» состоит в освоении студентами, получающими квалификацию бакалавра, подходов к составлению алгоритмов решения задач различной степени сложности, а также овладении навыками создания программ на примере среды разработки приложений Microsoft Visual Studio.

Задачами дисциплины являются: изучение основополагающих понятий и правил программирования; разработка алгоритмов обработки данных различной структуры, освоение правил создания и организации пользовательского интерфейса.

2. Место дисциплины в структуре образовательной программы

Учебная дисциплина «Программирование» относится к части, формируемой участниками образовательных отношений, ФГОС ВО по направлению подготовки 09.03.01 Информатика и вычислительная техника (квалификация «бакалавр»).

Для освоения дисциплины «Программирование» студент должен обладать знаниями и умениями, приобретенными в результате освоения предшествующих дисциплин:

- основы алгоритмизации и программирования.

Дисциплина «Программирование» является предшествующей и необходимой для изучения следующих дисциплин:

- объектно-ориентированное программирование;
- web-программирование.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Таблица 1

Код по ФГОС	Индикаторы достижения	Планируемые результаты обучения по дисциплине (ЗУВ)
Общепрофессиональные		
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8.1. Применяет языки программирования и языки работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. ОПК-8.2. Программирует, выполняет отладку и тестирование прототипов программно-технических	Знать: основные этапы и принципы создания программного продукта. Уметь: разрабатывать и тестировать программные компоненты информационных систем. Владеть: приемами отладки приложений, поиска ошибок и обработки исключений.

	комплексов задач.	
ПК-1 Способен выполнять работы по проектированию программного обеспечения и систем, автоматизирующих задачи организационного управления и бизнес-процессы	ПК-1.1. Проводит анализ требований к программному обеспечению. ПК-1.2. Разрабатывает технические спецификации на программные компоненты и их взаимодействие. ПК-1.3. Осуществляет проектирование программного обеспечения.	Знать: принципы и шаблоны проектирования программного обеспечения. Уметь: выполнять проектирование компонентов программного обеспечения по заданным требованиям в рамках предметной области. Владеть: практическим опытом разработки компонентов программного обеспечения и интерфейсов.

4. Объем дисциплины и виды учебной работы

Таблица 2

Вид учебной работы	Всего часов/ зач. ед.		Семестры				
	ОФО	ЗФО	4	5	5	6	
			ОФО		ЗФО		
Контактная работа (всего)	99/2,7	16/0,5	48/1,2	51/1,5	8/0,25	8/0,25	
В том числе:							
Лекции	31/0,9	4/0,1	16/0,4	17/0,5	2/0,05	2/0,05	
Практические занятия	-	-	-	-	-	-	
Практическая подготовка	-	-	-	-	-	-	
Лабораторные занятия	66/1,8	12/0,4	32/0,8	34/1	6/0,2	6/0,2	
Самостоятельная работа (всего)	153/4,3	236/6,5	72/2	81/2,3	108/3	128/3,5	
В том числе:							
Курсовая работа (проект)	-	-	-	-	-	-	
Расчетно-графические работы	-	-	-	-	-	-	
ИТР	-	-	-	-	-	-	
Рефераты	-	-	-	-	-	-	
Доклады с презентациями	45/1,3	72/2	18/0,5	27/0,8	36/1	36/1	
<i>И (или) другие виды самостоятельной работы:</i>							
Подготовка к лабораторным работам	36/1	72/2	18/0,5	18/0,5	36/1	36/1	
Подготовка к практическим занятиям	-	-	-	-	-	-	
Подготовка к зачету	36/1	36/1	36/1	-	36/1	-	
Подготовка к экзамену	36/1	56/1,5	-	36/1	-	56/1,5	
Вид отчетности			зач.	экс.	зач.	экс.	
Общая трудоемкость дисциплины	ВСЕГО в часах	252	252	120	132	116	136
	ВСЕГО в зач. ед.	7	7	3,2	3,8	3,25	3,75

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Таблица 3

№ п/п	Наименование раздела дисциплины по семестрам	Часы лекционных занятий		Часы лабораторных занятий		Часы практических (семинарских) занятий		Всего часов	
		ОФО	ЗФО	ОФО	ЗФО	ОФО	ЗФО	ОФО	ЗФО
4 семестр									
1.	Обработка данных в C#	8	2	24	4	-	-	32	6
2.	Программные компоненты: разработка, отладка и тестирование	8	-	8	2	-	-	16	2
5 семестр									
3.	Программирование приложений на языке C#	5	-	18	4	-	-	23	4
4.	Работа с источниками данных	12	2	16	2	-	-	28	4

5.2. Лекционные занятия

Таблица 4

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	Обработка данных в C#	Работа с массивами в C#: объявление массивов, основные операции с массивами. Одномерные и двумерные массивы. Обработка строк текста в C#. Основные функции обработки строк. Основные функции обработки даты и времени в C#.
2.	Программные компоненты: разработка, отладка и тестирование	Понятие и назначение отладки программы. Виды ошибок в программе. Понятие трассировки. Точки останова. Ошибки времени выполнения программы. Обработка исключений. Инструкция try...catch. Понятие, назначение и виды библиотек в программировании. Понятие и механизм работы API-функций.
3.	Программирование приложений на языке C#	Модульное программирование: пользовательские процедуры и функции. Графика в программировании. Карандаш и кисть. Методы рисования графических примитивов: прямоугольников, эллипсов, многоугольников.
4.	Работа с источниками данных	Работа с файлами и каталогами в C#: основные методы. Работа с базами данных в C#. Технология ADO.NET: основные используемые объекты. Взаимодействие приложения на языке C# с MS Excel. Делегаты в C#.

5.3. Лабораторные занятия

Таблица 5

№ п/п	Наименование раздела дисциплины	Наименование лабораторных работ
1.	Обработка данных в C#	Массивы. Типичные операции при работе с массивами. Использование массивов для хранения входных данных и результатов вычислений. Обработка текстовой информации в C#. Функции обработки строк. Автоматизация операций по обработке текста. Работа с датами и временем.
2.	Программные компоненты: разработка, отладка и тестирование	Методы отладки приложения. Трассировка, точки останова. Обработка исключений, инструкция try...catch.
3.	Программирование приложений на языке C#	Модульное программирование. Разработка пользовательских функций и процедур. Графика: рисование в программировании. Создание графических примитивов. Разработка приложений для различных областей применения.
4.	Работа с источниками данных	Работа с файлами. Использование файлов для обмена данными с приложением. Работа с каталогами как объектами файловой системы. Связь приложений на C# с базами данных. Взаимодействие приложений на C# с базами данных с помощью запросов на языке SQL. Взаимодействие приложения на языке C# с MS Excel.

5.4. Практические (семинарские) занятия: нет

Таблица 6

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	-	-

6. Самостоятельная работа студентов по дисциплине

Способ организации самостоятельной работы: подготовка презентации на 12-15 слайдов с устным докладом по заданной тематике; разработка приложений по заданию преподавателя.

4 семестр

Тематика докладов с презентациями:

1. Сферы применения ЯП: программирования для локальных ПК, для мобильных устройств, для сети Интернет.
2. Современные среды программирования: сравнительный анализ.
3. Язык программирования C++: история возникновения, особенности, наследие.
4. Язык программирования C#: история возникновения, особенности, перспективы.

5. Язык программирования Java: история возникновения, характерные особенности, сферы применения.
6. Логическая парадигма программирования: основные понятия, характерные особенности, области применения, достоинства и недостатки.
7. Функциональная парадигма программирования: основные понятия, характерные особенности, области применения, достоинства и недостатки.
8. Сравнительный анализ функционального, логического и процедурного подхода в программировании.
9. Понятие и назначение библиотек в программировании.
10. Парадигма объектно-ориентированного программирования: основные понятия, характерные особенности, области применения, достоинства и недостатки.
11. Инкапсуляция, наследование и полиморфизм в ООП.
12. Среда программирования Visual Studio.
13. Основные принципы технологии Microsoft .NET.
14. Методы тестирования программного обеспечения. Классификация ошибок в программном обеспечении
15. Правила и подходы к построению интерфейса пользователя.

5 семестр

Тематика докладов с презентациями:

1. Создание веб-страниц и программирование для Internet.
 2. Язык разметки веб-страниц HTML.
 3. Язык PHP: особенности, характеристики и сферы применения.
 4. Язык XML и обширные возможности его применения. Стандарты XML.
 5. ОС Android: история возникновения, особенности развития. Разработка приложений для Android.
 6. Язык запросов SQL: стандарты, функции, особенности. Связь и взаимодействие приложений с базами данных.
 7. Приемы отладки программ и обработки исключений. Оптимизация приложений.
 8. Групповая разработка проекта.
 9. Создание инсталляционного пакета приложения.
 10. Управление данными. Технология ADO.NET.
 11. Создание интернет-приложений. Технология ASP.NET.
 12. Классы в ООП: поля, методы и свойства.
 13. Создание и удаление классов и экземпляров классов.
 14. Понятие и состав интегрированной среды разработки приложений (IDE).
- Примеры популярных сред разработки.
15. Краткая характеристика основных современных сред разработки приложений.

Примеры заданий

Программа-лотерея. Разработать игру, в которой компьютер генерирует случайное число в диапазоне от 0 до 100. Четыре игрока делают ставки, пытаясь угадать число. Выигрывает тот, чья ставка оказалась ближе всего к «загаданному» компьютером числу.

Программа «Тренажер памяти». Компьютер генерирует случайную последовательность из 7 цифр. Игроку дается 4 секунды на запоминание, затем компьютер скрывает последовательность. Задача игрока – воспроизвести комбинацию

цифр. Игра включает 10 заданий, возможно усложнение условий.

Учебно-методическое обеспечение для самостоятельной работы студентов:

1. Котов, О.М. Язык C#. Краткое описание и введение в технологии программирования [Электронный ресурс]: учебное пособие / О.М. Котов. — Электрон. текстовые данные. – Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2014. – 208 с. – Режим доступа: <http://www.iprbookshop.ru/68524.html> (ЭБС «IPRbooks»).

2. Биллиг, В.А. Основы программирования на C# [Электронный ресурс] / В. А. Биллиг. – 2-е изд. – Электрон. текстовые данные. – М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 574 с. – Режим доступа: <http://www.iprbookshop.ru/73695.html> (ЭБС «IPRbooks»).

7. Оценочные средства

7.1. Вопросы к рубежным аттестациям

4 семестр

К 1-ой рубежной аттестации:

1. Работа с массивами в C#: объявление массивов, основные операции с массивами. Одномерные и двумерные массивы.

2. Обработка строк текста в C#. Основные функции обработки строк: Length, IndexOf, Trim, Remove, ToUpper.

3. Обработка строк текста в C#. Основные функции обработки строк: LastIndexOf, Substring, Insert, Replace, ToLower.

4. Основные функции обработки даты и времени в C#.

Ко 2-ой рубежной аттестации:

1. Понятие и назначение отладки программы. Виды ошибок в программе. Понятие трассировки. Точки останова.

2. Ошибки времени выполнения программы. Обработка исключений. Инструкция try...catch.

3. Понятие, назначение и виды библиотек в программировании.

4. Понятие и механизм работы API-функций.

5 семестр

К 1-ой рубежной аттестации:

1. Модульное программирование: пользовательские процедуры и функции в C#.

2. Графика в программировании. Карандаш и кисть.

3. Методы рисования графических примитивов: прямоугольников, эллипсов, многоугольников.

4. Основные методы работы с файлами. Использование файлов для обмена данными с приложением.

5. Работа с каталогами как объектами файловой системы: основные методы.

Ко 2-ой рубежной аттестации:

1. Работа с базами данных в C#.

2. Технология ADO.NET: основные используемые объекты.

3. Взаимодействие приложения на языке C# с MS Excel.

4. Понятие и состав интегрированной среды разработки приложений (IDE).

Примеры популярных сред разработки.

5. Краткая характеристика основных современных сред разработки приложений.
6. Делегаты в С#.

Помимо проверки знания теоретического материала, на аттестации / экзамене студентам предлагаются практические задания по разделам дисциплины.

Образец билетов рубежной аттестации:

<p style="text-align: center;">Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Программирование» 1-я рубежная аттестация</p> <p style="text-align: center;">Группа: _____ Семестр: 4</p> <p style="text-align: center;">Билет №</p> <p>1. Работа с массивами в С#: объявление массивов, основные операции с массивами. Одномерные и двумерные массивы.</p> <p>2. Вводится массив из 6 целых чисел. Требуется вычислить произведение элементов, являющихся нечетными числами.</p> <p style="text-align: center;">Преподаватель _____</p>
--

<p style="text-align: center;">Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Программирование» 2-я рубежная аттестация</p> <p style="text-align: center;">Группа: _____ Семестр: 4</p> <p style="text-align: center;">Билет №</p> <p>1. Понятие и назначение отладки программы. Виды ошибок в программе. Понятие трассировки. Точки останова.</p> <p>2. Вводится слово. Определить его длину, вторую и предпоследнюю букву.</p> <p style="text-align: center;">Преподаватель _____</p>

<p style="text-align: center;">Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Программирование» 1-я рубежная аттестация</p> <p style="text-align: center;">Группа: _____ Семестр: 5</p> <p style="text-align: center;">Билет №</p> <p>1. Работа с каталогами как объектами файловой системы: основные методы.</p> <p>2. Вводятся три числа. Программа должна указать, является ли их сумма двузначным числом.</p> <p style="text-align: center;">Преподаватель _____</p>
--

Грозненский Государственный Нефтяной Технический Университет
им. акад. М.Д. Миллионщикова
Кафедра «Информационные технологии»
Дисциплина «Программирование»
2-я рубежная аттестация

Группа:

Семестр: 5

Билет №

1. Работа с базами данных в C#.
2. Пользователь вводит числа А и В. Найти сумму и произведение чисел в диапазоне от А до В.

Преподаватель _____

7.2. Вопросы к зачету / экзамену

4 семестр

Вопросы к зачету

1. Работа с массивами в C#: объявление массивов, основные операции с массивами. Одномерные и двумерные массивы.
2. Обработка строк текста в C#. Основные функции обработки строк: Length, IndexOf, Trim, Remove, ToUpper.
3. Обработка строк текста в C#. Основные функции обработки строк: LastIndexOf, Substring, Insert, Replace, ToLower.
4. Основные функции обработки даты и времени в C#.
5. Понятие и назначение отладки программы. Виды ошибок в программе. Понятие трассировки. Точки останова.
6. Ошибки времени выполнения программы. Обработка исключений. Инструкция try...catch.
7. Понятие, назначение и виды библиотек в программировании.
8. Понятие и механизм работы API-функций.

5 семестр

Вопросы к экзамену

1. Модульное программирование: пользовательские процедуры и функции в C#.
2. Графика в программировании. Карандаш и кисть.
3. Методы рисования графических примитивов: прямоугольников, эллипсов, многоугольников.
4. Основные методы работы с файлами. Использование файлов для обмена данными с приложением.
5. Работа с каталогами как объектами файловой системы: основные методы.
6. Работа с базами данных в C#.
7. Технология ADO.NET: основные используемые объекты.
8. Взаимодействие приложения на языке C# с MS Excel.
9. Понятие и состав интегрированной среды разработки приложений (IDE).
Примеры популярных сред разработки.
10. Краткая характеристика основных современных сред разработки приложений.
11. Делегаты в C#.

Образец билета к зачету:

Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Программирование»	
Группа:	Семестр: 4
Билет №	
1. Ошибки времени выполнения программы. Обработка исключений. Инструкция try...catch.	
2. Вводится массив из 6 целых чисел. Требуется вычислить произведение элементов, являющихся нечетными числами.	
Подпись преподавателя _____	Подпись заведующего кафедрой _____

Образец билета к экзамену:

Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Программирование»	
Группа:	Семестр: 5
Билет №	
1. Модульное программирование: пользовательские процедуры и функции в C#.	
2. Пользователь вводит два положительных числа: M и N. Найти произведение чисел в диапазоне от M до N, исключив сами значения M и N.	
Подпись преподавателя _____	Подпись заведующего кафедрой _____

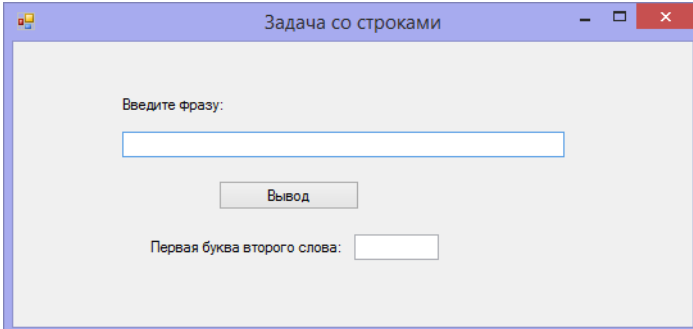
7.3. Текущий контроль

Образец типового задания для лабораторных занятий

Лабораторная работа на тему «Обработка текстовой информации в C#.

Функции обработки строк»

Пользователь вводит любую фразу. Программа должна выделить и вывести первую букву второго слова.



Программный код

```
private void button1_Click(object sender, EventArgs e)
{
```

str – введенная фраза, s – искомая буква; n – позиция пробела в строке как разделителя слов.

```
    string str, s;
    int n;
    str = textBox1.Text;
```

Значение n находится с помощью метода IndexOf, определяющего индекс с отсчетом от нуля первого вхождения указанного символа в данной строке.

```
    n = str.IndexOf(" ");
```

С помощью метода Substring выделяется 1 символ с позиции, следующей сразу за пробелом (n+1).

```
s = str.Substring(n + 1, 1);  
textBox2.Text = s;  
}
```

Для самостоятельного выполнения

Выделить из введенной пользователем фразы последнее слово в отдельное текстовое поле.

7.4. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

Таблица 7

Планируемые результаты освоения компетенции	Критерии оценивания результатов обучения				Наименование оценочного средства
	менее 41 баллов (неудовлетворительно)	41-60 баллов (удовлетворительно)	61-80 баллов (хорошо)	81-100 баллов (отлично)	
ОПК-8: Способность разрабатывать алгоритмы и программы, пригодные для практического применения					
Знать: основные этапы и принципы создания программного продукта.	Фрагментарные знания	Неполные знания	Сформированные, но содержащие отдельные пробелы знания	Сформированные систематические знания	Комплект заданий для выполнения лабораторных работ, темы докладов с презентациями, вопросы по темам / разделам дисциплины
Уметь: разрабатывать и тестировать программные компоненты информационных систем.	Частичные умения	Неполные умения	Умения полные, допускаются небольшие ошибки	Сформированные умения	
Владеть: приемами отладки приложений, поиска ошибок и обработки исключений.	Частичное владение навыками	Несистематическое применение навыков	В систематическом применении навыков допускаются пробелы	Успешное и систематическое применение навыков	
ПК-1: Способность выполнять работы по проектированию программного обеспечения и систем, автоматизирующих задачи организационного управления и бизнес-процессы					
Знать: принципы и шаблоны проектирования программного обеспечения.	Фрагментарные знания	Неполные знания	Сформированные, но содержащие отдельные пробелы знания	Сформированные систематические знания	Комплект заданий для выполнения лабораторных работ, темы докладов с презентациями, вопросы по темам / разделам дисциплины
Уметь: выполнять проектирование компонентов программного обеспечения по заданным требованиям в рамках предметной области.	Частичные умения	Неполные умения	Умения полные, допускаются небольшие ошибки	Сформированные умения	

Владеть: практическим опытом разработки компонентов программного обеспечения и интерфейсов.	Частичное владение навыками	Несистематическое применение навыков	В систематическом применении навыков допускаются пробелы	Успешное и систематическое применение навыков
--	-----------------------------	--------------------------------------	--	---

8. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации обучающихся созданы фонды оценочных средств, адаптированные для инвалидов и лиц с ограниченными возможностями здоровья и позволяющие оценить достижение ими запланированных в основной образовательной программе результатов обучения и уровень сформированности всех компетенций, заявленных в образовательной программе. Форма проведения текущей аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.). При тестировании для слабовидящих студентов используются фонды оценочных средств с укрупненным шрифтом. На экзамен приглашается сопровождающий, который обеспечивает техническое сопровождение студенту. При необходимости студенту-инвалиду предоставляется дополнительное время для подготовки ответа на экзамене (или зачете). Обучающиеся с ограниченными возможностями здоровья и обучающиеся инвалиды обеспечиваются печатными и электронными образовательными ресурсами (программы, учебные пособия для самостоятельной работы и т.д.) в формах, адаптированных к ограничениям их здоровья и восприятия информации:

1) для инвалидов и лиц с ограниченными возможностями здоровья **по зрению:**

- **для слепых:** задания для выполнения на семинарах и практических занятиях оформляются рельефно-точечным шрифтом Брайля или в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением для слепых, либо зачитываются ассистентом; письменные задания выполняются на бумаге рельефно-точечным шрифтом Брайля или на компьютере со специализированным программным обеспечением для слепых либо надиктовываются ассистенту; обучающимся для выполнения задания при необходимости предоставляется комплект письменных принадлежностей и бумага для письма рельефно-точечным шрифтом Брайля, компьютер со специализированным программным обеспечением для слепых;

- **для слабовидящих:** обеспечивается индивидуальное равномерное освещение не менее 300 люкс; обучающимся для выполнения задания при необходимости предоставляется увеличивающее устройство; возможно также использование собственных увеличивающих устройств; задания для выполнения заданий оформляются увеличенным шрифтом;

2) для инвалидов и лиц с ограниченными возможностями здоровья **по слуху:**

- **для глухих и слабослышащих:** обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающимся предоставляется звукоусиливающая аппаратура индивидуального пользования; предоставляются услуги сурдопереводчика;

- **для слепоглухих** допускается присутствие ассистента, оказывающего услуги тифлосурдопереводчика (помимо требований, выполняемых соответственно для слепых и глухих);

3) для лиц с тяжелыми нарушениями речи, глухих, слабослышащих лекции и семинары, проводимые в устной форме, проводятся в письменной форме;

4) для инвалидов и лиц с ограниченными возможностями здоровья, **имеющих**

нарушения опорно-двигательного аппарата:

- для лиц с нарушениями опорно-двигательного аппарата, нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей: письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту; выполнение заданий (тестов, контрольных работ), проводимые в письменной форме, проводятся в устной форме путем опроса, беседы с обучающимся.

9. Учебно-методическое и информационное обеспечение дисциплины

1. Агапов, В.П. Основы программирования на языке С# [Электронный ресурс]: учебное пособие / В.П. Агапов. – Электрон. текстовые данные. – М.: Московский государственный строительный университет, ЭБС АСВ, 2012. – 128 с. – Режим доступа: <http://www.iprbookshop.ru/16366.html> (ЭБС «IPRbooks»).

2. Поляков, А.Ю. Программирование [Электронный ресурс]: практикум / А.Ю. Поляков, А.Ю. Полякова, Е.Н. Перишкова. – Электрон. текстовые данные. – Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2015. – 55 с. – Режим доступа: <http://www.iprbookshop.ru/55494.html> (ЭБС «IPRbooks»).

3. Котов, О.М. Язык С#. Краткое описание и введение в технологии программирования [Электронный ресурс]: учебное пособие / О.М. Котов. — Электрон. текстовые данные. – Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2014. – 208 с. – Режим доступа: <http://www.iprbookshop.ru/68524.html> (ЭБС «IPRbooks»).

4. Биллиг, В.А. Основы программирования на С# [Электронный ресурс] / В. А. Биллиг. – 2-е изд. – Электрон. текстовые данные. – М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 574 с. – Режим доступа: <http://www.iprbookshop.ru/73695.html> (ЭБС «IPRbooks»).

10. Материально-техническое обеспечение дисциплины

10.1. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Перечень материально-технических средств учебной аудитории для проведения занятий по дисциплине:

- учебная аудитория, доска;
- стационарные компьютеры;
- мультимедийный проектор;
- настенный экран.

10.2. Помещения для самостоятельной работы

Учебная аудитория для самостоятельной работы – 4-01.

Методические указания по освоению дисциплины «Программирование»

1. Методические указания для обучающихся по планированию и организации времени, необходимого для освоения дисциплины

Изучение рекомендуется начать с ознакомления с рабочей программой дисциплины, ее структурой и содержанием разделов (модулей), фондом оценочных средств, ознакомиться с учебно-методическим и информационным обеспечением дисциплины.

Дисциплина «Программирование» состоит из четырех связанных между собой разделов, обеспечивающих последовательное изучение материала.

Обучение по дисциплине «Программирование» осуществляется в следующих формах:

1. Аудиторные занятия (лекции, лабораторные занятия).
2. Самостоятельная работа студента (подготовка к лекциям, лабораторным занятиям, доклады с презентациями, индивидуальная консультация с преподавателем).

Учебный материал структурирован и изучение дисциплины производится в тематической последовательности. Каждому лабораторному занятию и самостоятельному изучению материала предшествует лекция по данной теме. Обучающиеся самостоятельно проводят предварительную подготовку к занятию, принимают активное и творческое участие в обсуждении теоретических вопросов, разборе проблемных ситуаций и поисков путей их решения.

Описание последовательности действий обучающегося:

При изучении курса следует внимательно слушать и конспектировать материал, излагаемый на аудиторных занятиях. Для его понимания и качественного усвоения рекомендуется следующая последовательность действий:

1. После окончания учебных занятий для закрепления материала просмотреть и обдумать текст лекции, прослушанной сегодня, разобрать рассмотренные примеры (10- 15 минут).
2. При подготовке к лекции следующего дня повторить текст предыдущей лекции, подумать о том, какая может быть следующая тема (10-15 минут).
3. В течение недели выбрать время для работы с литературой в электронной библиотечной системе (по 1 часу).
4. При подготовке к лабораторному занятию повторить основные понятия по теме, изучить примеры. Решая конкретную ситуацию, – предварительно понять, какой теоретический материал нужно использовать. Наметить план решения, попробовать на его основе решить 1-2 задачи.

2. Методические указания по работе обучающихся во время проведения лекций

Лекции дают обучающимся систематизированные знания по дисциплине, концентрируют их внимание на наиболее сложных и важных вопросах. Лекции обычно излагаются в традиционном или в проблемном стиле. Для студентов в большинстве случаев в проблемном стиле. Проблемный стиль позволяет стимулировать активную познавательную деятельность обучающихся и их интерес к дисциплине, формировать творческое мышление, прибегать к противопоставлениям и сравнениям, делать обобщения, активизировать внимание обучающихся путем постановки проблемных вопросов, поощрять дискуссию.

Во время лекционных занятий рекомендуется вести конспектирование учебного материала, обращать внимание на формулировки и категории, раскрывающие суть того или иного явления, выводы и практические рекомендации.

Работая над конспектом лекций, необходимо использовать также литературу,

которую дополнительно рекомендовал преподаватель. Тематика лекций дается в рабочей программе дисциплины.

3. Методические указания обучающимся по подготовке к лабораторным занятиям

На лабораторных занятиях приветствуется активное участие в обсуждении конкретных ситуаций, способность на основе полученных знаний находить наиболее эффективные решения задач, уметь находить полезный дополнительный материал по тематике занятий.

Студенту рекомендуется следующая схема подготовки к лабораторному занятию:

1. Ознакомиться с планом занятия, который отражает содержание предложенной темы.

2. Проработать конспект лекций.

3. Прочитать основную и дополнительную литературу.

В процессе подготовки к лабораторным занятиям, необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов отношение к конкретной проблеме.

4. Выполнить домашнее задание.

5. При затруднениях сформулировать вопросы к преподавателю.

Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы, выступать и участвовать в коллективном обсуждении вопросов изучаемой темы, правильно выполнять практические задания, которые даются в фонде оценочных средств дисциплины.

4. Методические указания обучающимся по организации самостоятельной работы

Цель организации самостоятельной работы по дисциплине «Программирование» – это углубление и расширение знаний в области программирования; формирование навыка и интереса к самостоятельной познавательной деятельности.

Самостоятельная работа обучающихся является важнейшим видом освоения содержания дисциплины, подготовки к практическим занятиям и к рубежной аттестации. Самостоятельная работа носит исследовательский характер, что послужит в будущем основанием для написания выпускной квалификационной работы, практического применения полученных знаний.

Организация самостоятельной работы обучающихся ориентируется на активные методы овладения знаниями, развитие творческих способностей, переход от поточного к индивидуализированному обучению, с учетом потребностей и возможностей личности.

Правильная организация самостоятельных учебных занятий, их систематичность, целесообразное планирование рабочего времени позволяет студентам развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивать высокий уровень успеваемости в период обучения, получить навыки повышения профессионального уровня.

Подготовка к лабораторному занятию включает, кроме проработки конспекта и презентации лекции, поиск литературы (по рекомендованным спискам и самостоятельно).

При подготовке к контрольной работе (рубежной аттестации) обучающийся должен повторять пройденный материал в строгом соответствии с учебной программой, используя конспект лекций и литературу, рекомендованную преподавателем. При необходимости можно обратиться за консультацией и методической помощью к преподавателю.

Самостоятельная работа реализуется:

- непосредственно в процессе аудиторных занятий – на лекциях, лабораторных занятиях;
- в контакте с преподавателем вне рамок расписания – на консультациях по учебным вопросам, в ходе творческих контактов, при ликвидации задолженностей, при выполнении индивидуальных заданий и т.д.
- в библиотеке, дома, на кафедре при выполнении обучающимся учебных и практических задач.

Виды СРС и критерии оценок

(по балльно-рейтинговой системе ГГНТУ, СРС оценивается в 15 баллов)

1. Доклад с презентацией
2. Подготовка к лабораторным занятиям

Темы для самостоятельной работы прописаны в рабочей программе дисциплины. Эффективным средством осуществления обучающимся самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к учебным планам, рабочим программам дисциплин (модулей), практик, к изданиям электронных библиотечных систем.

Составитель:

Старший преподаватель кафедры
«Информационные технологии»



/ Д.А.Мачуева /

СОГЛАСОВАНО:

Зав. кафедрой
«Информационные технологии»



/ Н.А. Моисеенко /

Зав. выпускающей кафедрой
«Информатика и вычислительная техника»



/ Э. Д. Алисултанова /

Директор ДУМР



/ М.А. Магомаева /